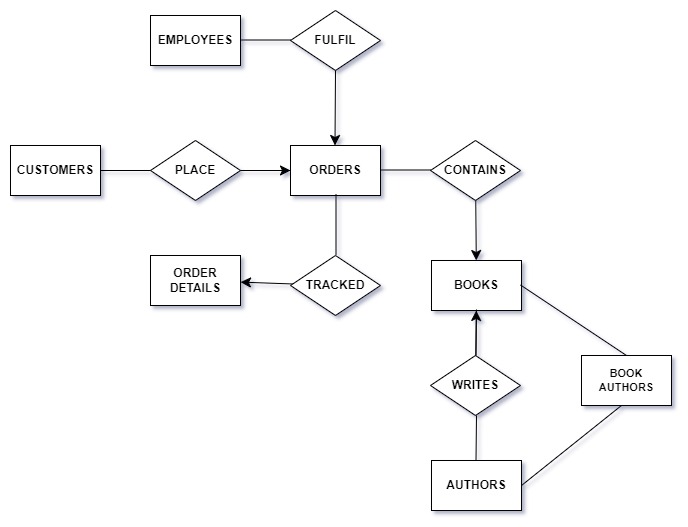
**XYLEMBD Online bookstore database management model design**

**Conceptual Data Model:**

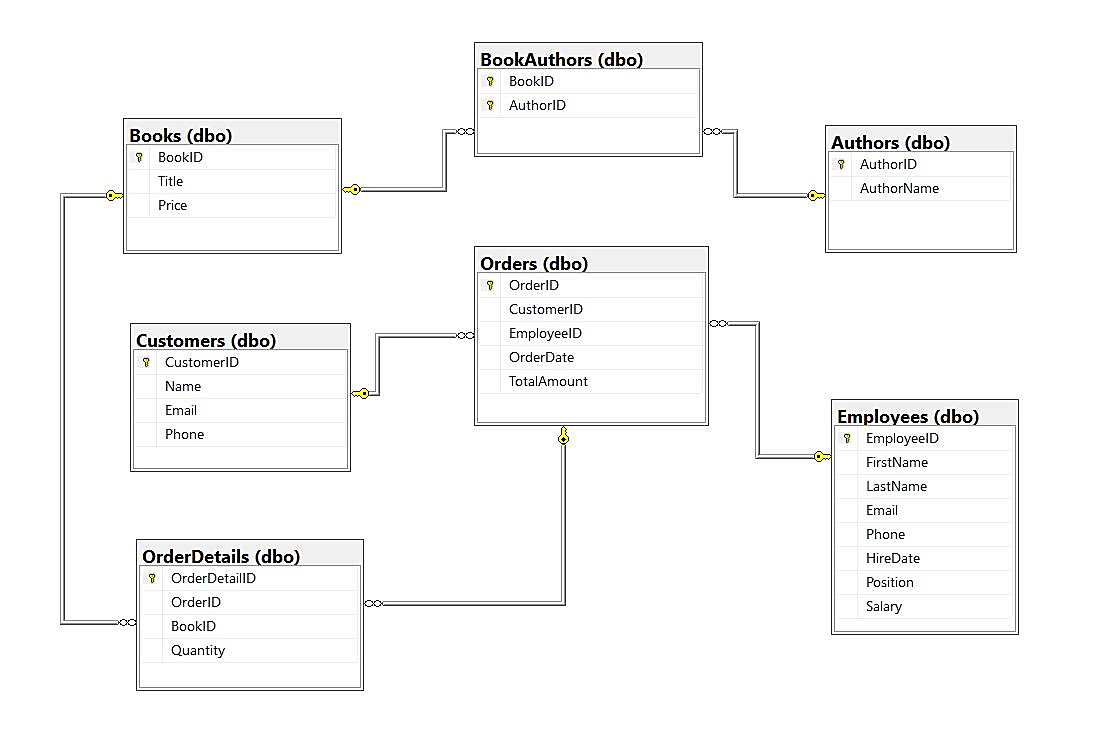
|  |  |  |
| --- | --- | --- |
| S/N | ENTITY | ATTRIBUTES |
| 1 | Customer | CustomerID, Name, Phone, Email |
| 2 | Orders | OrderID, CustomerID, OrderDate, TotalAmount, EmployeeID |
| 3 | OrderDetails | OrderDetailID(PK), OrderID(FK), BookID(FK), Quantity |
| 4 | Books | BookID(PK), Title, Price |
| 5 | Authors | AuthorID(PK), Name |
| 6 | BookAuthors | BookID(FK), AuthorID(FK) |
| 7 | Employees | EmployeeID, FirstName, LastName, Position, HireDate, Salary, Phone, Email |





**Logical Data Model:**

|  |  |  |
| --- | --- | --- |
| S/N | ENTITY | ATTRIBUTES |
| 1 | Customer | CustomerID INT (PK) Identity 1,1) NOT NULL,  Name NVARCHAR (100) NOT NULL,  Phone INT NOT NULL,  Email NVARCAHR (100) NOT NULL |
| 2 | Orders | OrderID INT (PK) Identity (1, 1) NOT NULL,  CustomerID INT FOREIGN KEY REFERENCES Customers (CustomerID),  OrderDate DATE NOT NULL,  TotalAmount DECIMAL (10, 2) |
| 3 | Books | BookID INT (PK) Identity (1, 1) NOT NULL,  Title NVARCAHR (200) NOT NULL,  Price DECIMAL (10, 2) |
| 4 | Authors | MenuID INT (PK),  Name NVARCHAR (100),  Description NVARCAHR (100), |
| 5 | BookAuthors | BookID FOREIGH KEY REFERENCES Books (BookID)  AuthorID FOREIGH KEY REFERENCES Authors (AuthorID) |
| 6 | OrderDetails | OrderDetailID INT (PK) Identity (1, 1) NOT NULL,  OrderID INT FOREIGN KEY REFERENCES Orders (OrderID),  DeliveryDate DATE,  Status BOOLEAN |
| 7 | Employees | EmployeeID INT (PK) Identity (1, 1) NOT NULL,  First Name NVARCAHR (100),  Last Name NVARCAHR (100),  Phone NVARCAHR (100) NOT NULL,  Email NVARCHAR (100) NOT NULL  Position NVARCHAR (100),  Salary DECIMAL (10, 2),  HireDate DATE |



**Figure 2:ER Diagram**

**Entity-Relationship Diagram (ERD) Explanation:**

In the Xylem database, the relationships between tables are established through primary and foreign key constraints. Here's a brief explanation of the relationships based on the provided table descriptions:

1. Customers and Orders: This is a one-to-many relationship, where one customer can place multiple orders. The Orders table likely has a foreign key column referencing the CustomerID column in the Customers table.

2. Orders and OrderDetails: This is also a one-to-many relationship, where one order can have multiple order details. The OrderDetails table likely has a foreign key column referencing the OrderID column in the Orders table.

3. Books and OrderDetails: This is a many-to-many relationship, where multiple books can appear in multiple orders, and each order can contain multiple books. To represent this relationship, there is typically a junction table (not explicitly mentioned) that connects the Books and OrderDetails tables. This junction table would likely contain foreign key columns referencing both the BookID column in the Books table and the OrderID column in the OrderDetails table.

4. Authors and Books: This is also a many-to-many relationship, where multiple authors can write multiple books, and each book can have multiple authors. Similar to the relationship between Books and OrderDetails, there would likely be a junction table connecting the Authors and Books tables. This junction table would contain foreign key columns referencing both the AuthorID column in the Authors table and the BookID column in the Books table.

5. Employees and Orders: This is a one-to-many relationship, where one employee can handle multiple orders. The Orders table likely has a foreign key column referencing the EmployeeID column in the Employees table.

These relationships ensure data integrity and enable efficient querying and retrieval of information across related tables in the database.